Using Echoic Flow for the Guidance and Control of an Unmanned Aerial System

Undergraduate Thesis

Justin Kuric

Department of Electrical and Computer Engineering

The Ohio State University

Presented in partial fulfillment of the requirements for an Honors Research Distinction

Acknowledgments

I would like to thank Dr. Graeme Smith and Dr. Chris Baker. You made this project a valuable and enjoyable experience.

Table of Contents

List of Illustrations

Abstract – Bats are able to navigate through complex environments in low light conditions with enough precision to intercept insects in midair. It has been suggested that bats accomplish this task through echolocation and the use of echoic flow. At its simplest, echoic flow is a ratio of the distance from a target to the velocity towards the target. This parameter is known as τ, it is the time to collision with the target. This project explores the use of echoic flow for the guidance and control of an Unmanned Aerial System (UAS). The UAS platform used to implement this technique was a quadrotor outfitted with an acoustic range sensor. The objective was to have the UAS perform different braking strategies for landing through a perception of τ. Because errors in the measured distance were present during a maneuver, different strategies for filtering the distance measurements were developed. Polynomials at various orders were fit to the incoming measurements to determine the UAS's actual distance from the target as it moved. The combination of this filtering and the use of echoic flow allowed the UAS to control its approach to a target, for landing or otherwise, in a simple and accurate way.

I. Introduction

Society is finding an increasing number of applications for unmanned aerial systems (UAS). From repairing power lines [10], to becoming a personal photographer [9], to catching other UASs [8], these systems are being pushed into increasingly complex environments and are being asked to perform increasingly complex tasks. Autonomously guiding a UAS around obstacles like people and trees presents a difficult challenge. Fortunately a creature that has been doing this for millennia already exists in nature. The echolocating bat is a shining example of what UASs could potentially do. In low light conditions, bats are able fly in swarms of thousands, catch insects through the air, and land swiftly onto a branch. This research investigates using what is known about the flight patterns of bats for engineering better methods for controlling a UAS.

The researchers in [7], have concluded that bats use "acoustic flow fields" to navigate through their environment. Bats use the flow field parameter, τ, to compute flight paths around obstacles and toward targets. τ is the time until the bat collides with a target. When a bat is making an approach to intercept an insect or land on a branch, it will keep the derivative of τ at a constant value

throughout the motion. The objective of this research was to see if a UAS could also use this method for accurately controlling how it landed. Having explicit control of the path a UAS takes as it lands would be important for a UAS package delivery system. A UAS would be expected to land softly when delivering a package and land precisely enough to connect to a charging station after a delivery.

A landing method based on perceiving a UAS's τ with the ground was devised and then implemented on a UAS with an acoustic range sensor. Landing the UAS was a simple case for evaluating the method's effectiveness, but the same approach could also be used to control a UAS's approach toward any stationary obstacle. Also, since τ is calculated using the relative motions between objects, this method could be extended to control the UAS's approach towards a moving object.

The major challenge encountered in this research was accounting for errors in the range measurements and movements of the UAS. Small range measurement errors caused large errors in the perceived τ. Also, the UAS would not move at the exact velocity it was instructed to move at. To solve this issue, polynomials of various orders were fitted to the range measurements during a landing. These polynomials were then used to produce better estimates of the UAS's range based on the trend of the past range measurements. The majority of the research conducted here was dedicated to evaluating the effects and feasibility of this filtering technique because dealing with these errors was the key challenge for engineering this method for use in real situations.

II. Echoic Flow

2.1 Introduction to Echoic Flow

Acoustic or echoic flow is a specialization of General Tau Theory as it is described in [5]. The basic flow field parameter is τ and it is the ratio of the magnitude of a measurement to the rate of change of that measurement. Echoic flow deals with the measurements associated with active radar and sonar

platforms [6]. This means τ can be the ratio of the range measured by a radar platform to the rate of change of the range. This is expressed as

$$\tau_r = r/\dot{r} \qquad (1)$$

where r represents the range to a target and $\dot{r}$ is the change in r over time. Another way to think of $\dot{r}$ is as the platform's relative velocity. τ_r is the time until the range gap is closed, or in other words, the time until a collision occurs. For example, if a platform is moving toward a wall 4 meters away at 2 meters per second, τ_r is 2 seconds.

The derivative of τ_r is a dimensionless parameter that describes the nature of the collision between the platform and target. It is expressed as

$$\dot{\tau}_r = \frac{\partial}{\partial t}\,\tau_r = 1 - \,\tau_r\,\frac{\ddot{r}}{\dot{r}} \qquad (2)$$

where $\ddot{r}$ is the second derivative of position, which is the relative acceleration of the platform. When $\dot{\tau}_r$ is a constant between zero and one, the platform will slow down as it approaches the target. Figure 1 shows the resulting motions for a platform approaching a target with different constant values of $\dot{\tau}_r$. The platform starts 2.7 meters from with target with an initial velocity of 0.5 meters per second in these simulations. The dotted lines show the cases where $0 < \dot{\tau}_r \leq ½$. For these cases the platform decelerates rapidly at the beginning of the approach and then slowly drifts into the target. The solid lines show the cases where $½ < \dot{\tau}_r < 1$. For these cases the platform decelerates rapidly at a close proximity to the target.

The equation that describes the platform's position as a function of time in Figure 1 is

$$r = r_0\left(1 + \dot{\tau}_r\,\frac{r_0}{\dot{r}_0}t\right)^{1/\dot{\tau}_r} \qquad (3)$$

This equation is given in [6] and the r_0 and $\dot{r}_0$ terms are the starting position and velocity respectively. Note the power that the curve is raised is inversely proportional to $\dot{\tau}_r$. The selection of $\dot{\tau}_r$determines the order of the position versus time curve. For example, if $\dot{\tau}_r$ is equal to 0.50, the position will be a quadratic function of time. Because the value of $\dot{\tau}_r$ is selected at the start of an approach, the highest degree of the position curve will be known throughout the motion.

This fact was exploited in the design of a smoothing filter for the range measurements taken by the UAS, and will be discussed in section 3.4.

The goal was to have the UAS follow the motions in Figure 1 by selecting a value for $\dot{\tau}_r$ at the start of a landing maneuver. The curves in Figure 1 represent the ideal paths for the UAS to follow for a particular $\dot{\tau}_r$.

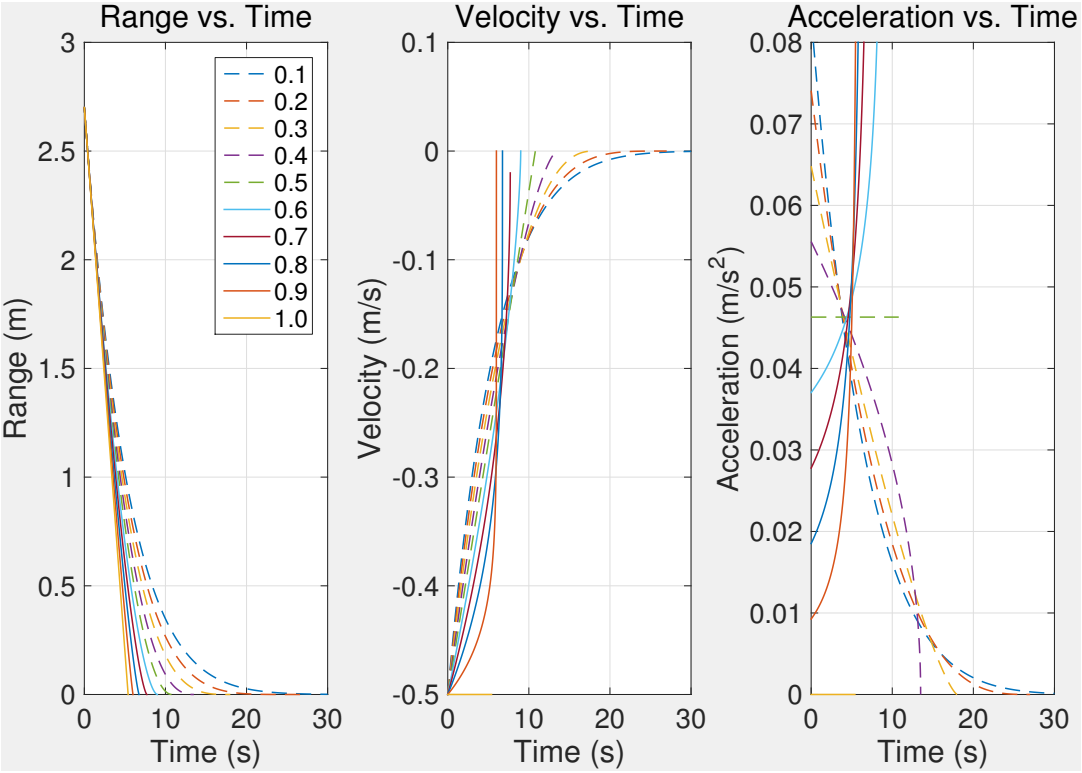

Figure 1: Platform Motion for Various $\dot{\tau}_r$ Values

2.2 Controlling a Vehicle with Echoic Flow

This section shows how the UAS attempted to keep $\dot{\tau}_r$ at a particular constant during a landing maneuver. This would allow the UAS to follow the ideal motions from Figure 1.

Height measurements had to be taken by the UAS at a periodic interval to perceive τ_r as it moved. After every measurement, the UAS would go through the process shown in Figure 2. As shown in the figure, each measurement was first passed through a smoothing filter. This filter helped obtain a better estimate of the UAS's true height. The velocity of the UAS was then calculated using the

current and previous height measurements. This is a linear approximation of the UAS's current velocity. Next τ_r was calculated. Then the acceleration needed to keep $\dot{\tau}_r$ at the specified constant was computed. The desired value of $\dot{\tau}_r$ would be input at the start of a landing maneuver. The equation used to find the acceleration was Equation 2 rearranged for $\ddot{r}$. The needed acceleration was then used to find the required velocity at the time of the next measurement. This assumed that setting the velocity would change result in a constant acceleration between the measurements. The UAS would then set its velocity to that value and repeat the process when a new height measurement was made

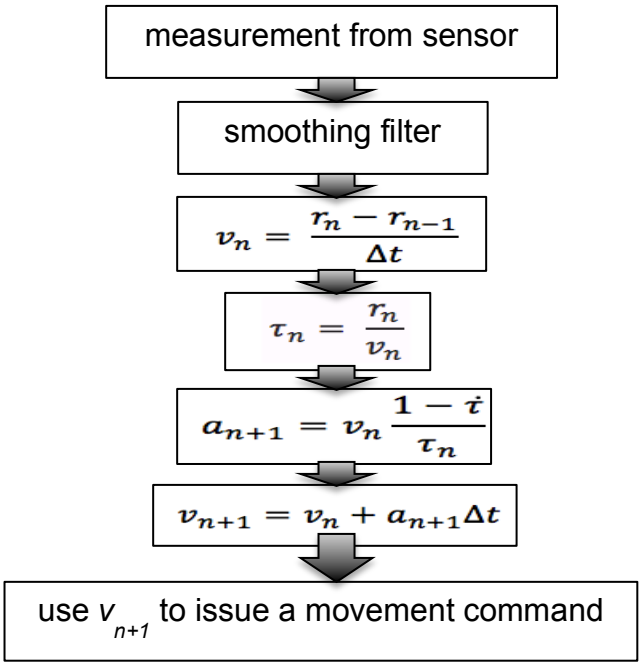

Figure 2: Control Flow of UAS During Landing

III. Equipment and Calibration

3.1 Overview of Equipment

The UAS used for this research was an AR Drone 2.0 equipped with an acoustic range sensor. A picture of the UAS is shown in Figure 3. The UAS communicated with a host computer over a Wi-Fi network. The host computer

sent commands over this network that set the UAS's velocity. The UAS also used this network to send information about its height and orientation back to the host computer in real time.

The acoustic range sensor was located on the bottom of the UAS and it was used to measure the UAS's height. The sensor was ultrasonic with an operating frequency of 25kHz. The sensor took measurements 15 times a second and would measure a number to three decimal places that corresponded to the UAS's height.

Figure 3 shows the experimental setup. The UAS would begin an experiment by flying to a specified height. It would then approach the ground with a constant velocity. For all actual and simulated experiments, the initial height was 2.7m and the initial velocity was 0.5m/s. Once the target velocity was reached, the UAS would control the landing by keeping $\dot{\tau}_r$ constant. The value of $\dot{\tau}_r$ was selected at the start of an experiment. The selection of $\dot{\tau}_r$, the initial velocity, and the initial height, determined the UAS's subsequent motion.

Figure 3: Experimental Setup

3.2 Calibration of the Range Sensor

To perform the experiment outlined in the previous section, the first step was to calibrate the acoustic range sensor onboard the UAS. The objective of

this calibration activity was to convert sensor's measurements to meters and to find the variance of those measurements. The variance would later be used in simulations of the UAS performing a landing.

To complete this objective, the UAS was held stationary above a large board. The distance between the board and the UAS was measured using a meter stick. The UAS was then turned on and the measurements from the acoustic sensor were collected on the host computer. The board was then moved to a different height and the process was repeated. Thus, the acoustic sensor's distance measurements could be compared to the measurements done with a meter stick. Figure 4 shows a set of measurements taken by the acoustic sensor with the board placed 938mm away. The red line on the figure shows the average sensor measurement during the period of time the UAS was above the board. Samples before and after the red line are when the board was being moved from underneath the UAS. The x-axis shows the sample number and the y-axis is the height measurement for that sample. The average was taken as the true sensor measurement for that distance.

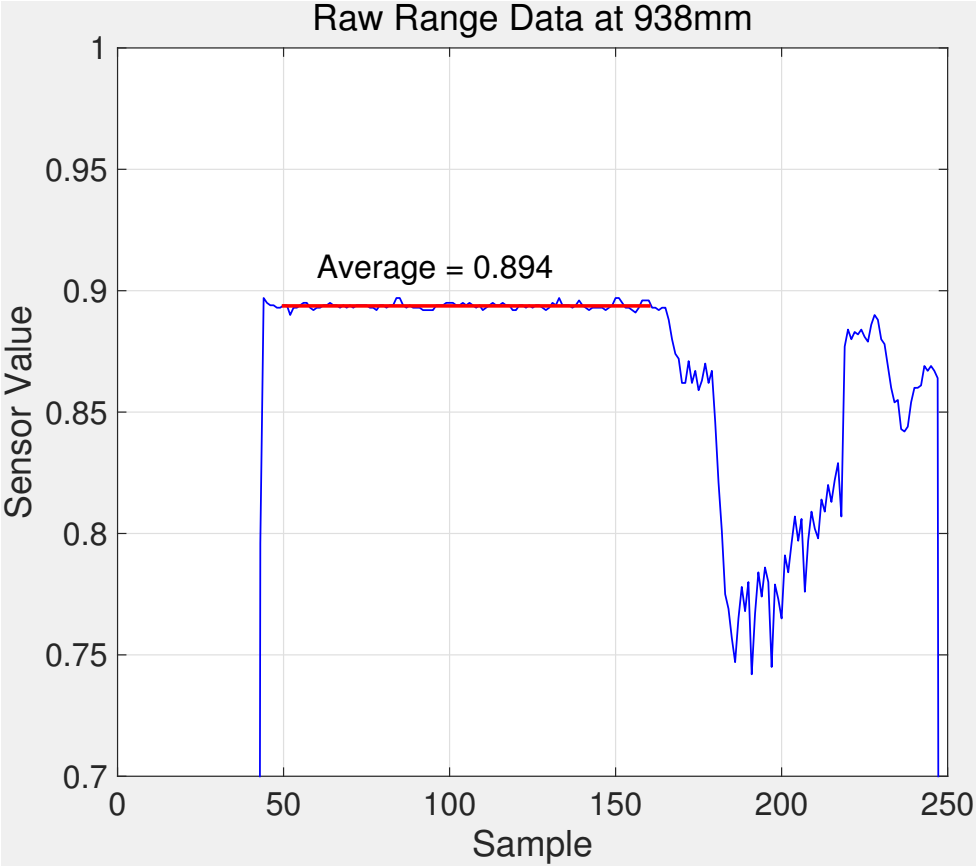

Figure 4: Range Measurements at a Fixed Height

The board was placed at distances of 938mm, 811mm, 734mm, 655mm, and 501mm. Figure 5 is a plot of the actual distance in meters versus what was measured from the sensor. The standard deviation of the sensor measurements was used to generate the error bars in Figure 5. A linear curve was fit to this data. The slope of the curve was close to one, meaning that the sensor's distance scale was in meters. However, the sensor had a 5 cm offset in its measurements. While this constant offset affected the true starting and stopping height of the UAS during an experiment, it did not affect any of the calculations that kept $\dot{t}_r$ at a constant during a landing.

The standard deviation at 501mm was 7mm. This was the maximum standard deviation seen in the measurements and was used to estimate measurement errors in simulations of the UAS's landing. The linear curve does not pass through the error bars in Figure 5 because they only show the

uncertainty of the sensor measurements. The actual uncertainty was larger due to errors in using the meter stick. But, finding the accuracy of the sensor was the objective of this calibration activity.

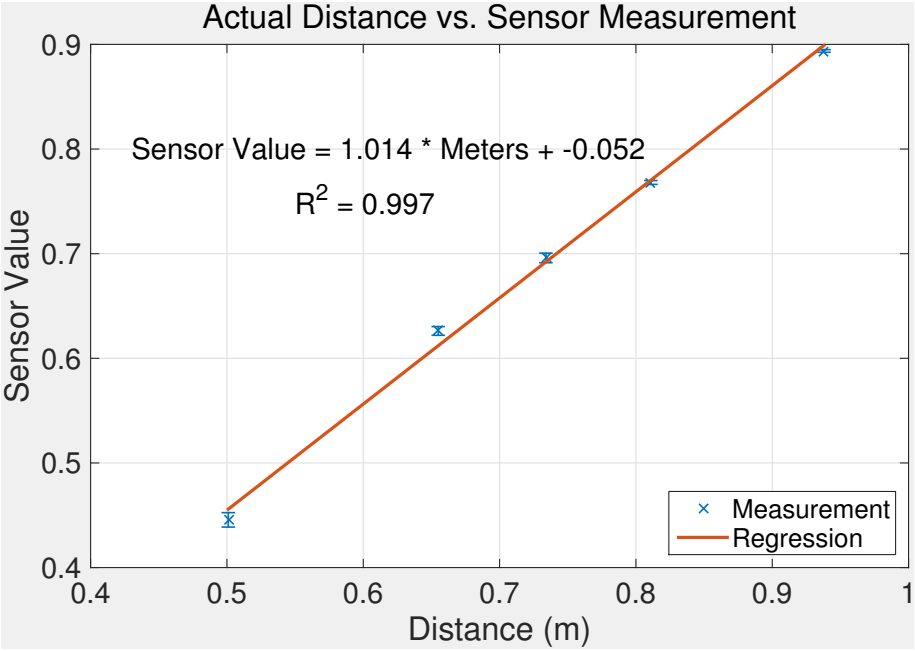

Figure 5: Calibration of the UAS's Acoustic Range Sensor

3.3 Calibration of the Velocity Control

When controlling the UAS from the host computer, a value between zero and one was sent over a Wi-Fi network to the UAS. The UAS set its downward velocity according to the magnitude of this number. The number corresponded to a percentage of the UAS's maximum speed. For example, if a value of 0.45 was sent, the UAS moved downward at 45% of its maximum downward velocity. The objective of the velocity calibration activity was to convert that percentage into units of meters per second and determine how accurately the UAS's velocity could be set.

To accomplish the calibration, the UAS flew to a height of two meters. It was then given the command to fly downward at certain percentage of its maximum speed. As it approached the ground, the UAS height sensor took height measurements. From these measurements, a height verses time plot was

generated. One of these plots is shown in Figure 6. The blue x's are individual measurements and the red line is a linear curve that was fit to those measurements. The slope of a linear curve was taken as the velocity of the UAS. The fit did not include the first ten data points because the UAS was accelerating during this period. The last few data points were also not included because updrafts formed when the UAS was close to the ground, creating nonlinearities in the UAS's movements.

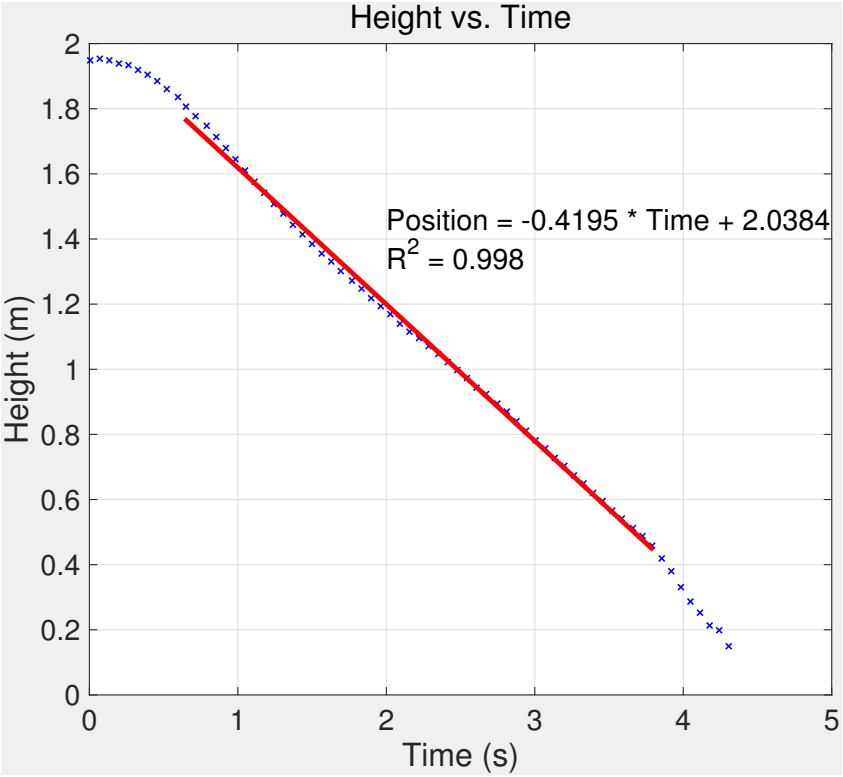

Figure 6: UAS Velocity Measurement

The UAS was given ten different movement commands. The first command was to approach the ground at 10% of its maximum downward velocity. This velocity was increased by 10% until the UAS was moving downward at 100% of its maximum velocity. The velocity was measured ten times for each command and the average of the set was used as the true velocity measurement. Figure 7 shows the average velocity of the UAS versus the percentage of its maximum speed it was set to. A quadratic curve was fit to this

data. This curve was used in subsequent experiments for having explicit control of the UAS's downward velocity.

The error bars in Figure 7 are generated from the standard deviation in the ten velocity measurements for each command. If the UAS was moving at less than 0.5 m/s, the maximum standard deviation was 12.6 mm/s. This value was used in simulations of the UAS landing. The UAS was kept below 0.5m/s in all of the experiments.

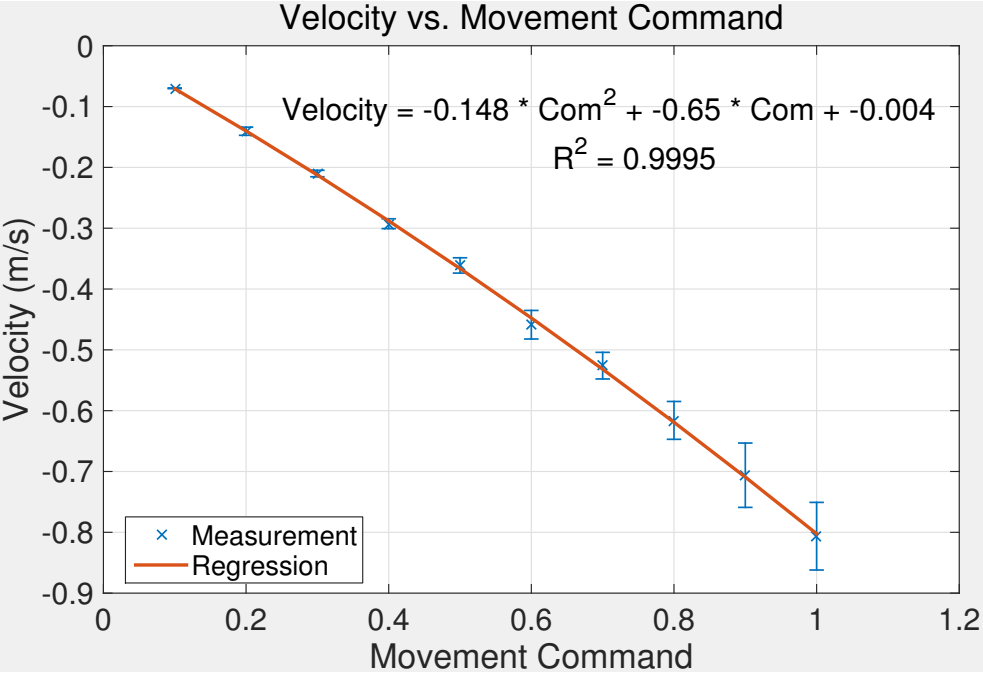

Figure 7: Calibration of the UAS's Velocity

3.5 Smoothing the Data

After a height measurement was taken, it passed through a smoothing filter to help mitigate errors in the range sensor's measurements. The smoothing filter used the previous height measurements and knowledge of the UAS's desired motion to produce a better estimate the current height. Because the order of the UAS's motion was known from Equation 3, a polynomial of that order could be fit to the height measurements to smooth the measurements during an

approach. For example, if $\dot{\tau}_r$ was 0.50, the resulting motion was quadratic and quadratic polynomials could be fitted to the height measurements.

The first step of the filtering algorithm was to store the current height and several past height measurements in a buffer. The size of the buffer could be altered to change the filter's properties. A polynomial was then fitted to the measurements in the buffer. The order of the polynomial could be selected to change the filter's properties. The polynomial was generated based on minimizing the squared error with the measurements in the buffer. The smoothed height estimate was obtained by evaluating the polynomial at the current time.

Figure 8 illustrates this process using a linear polynomial with a buffer size of four. This figure is only meant to show how the filter algorithm works and is not representative of the UAS performing a landing. A linear function was fitted to the current height and the previous three measurements. This curve was evaluated at the current time to obtain a new height estimate as shown by the red 'x'. The process was then repeated for each new measurement taken.

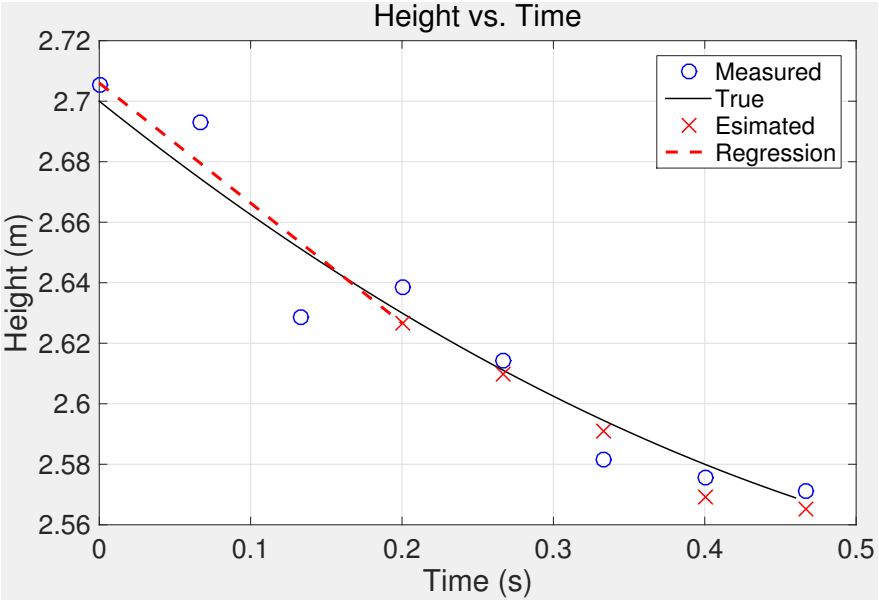

Figure 8: Simulated Size 4 Linear Filter

IV. Echoic Flow to Control UAS Landing

4.1 Simulations

MATLAB simulations of the UAS approaching the ground were performed to determine the optimum filter order and buffer size. An example simulation is shown in Figure 9. The height of the UAS progressed through time according to the control structure in Figure 2. Pseudorandom numbers with a Gaussian distribution were added to the height measurements and velocity instructions. The Gaussian distributions of these numbers had standard deviations according to what was predicted during the calibration activities. The simulated flight path was then compared with the ideal path from Figure 1 by calculating the root mean squared error between the two curves.

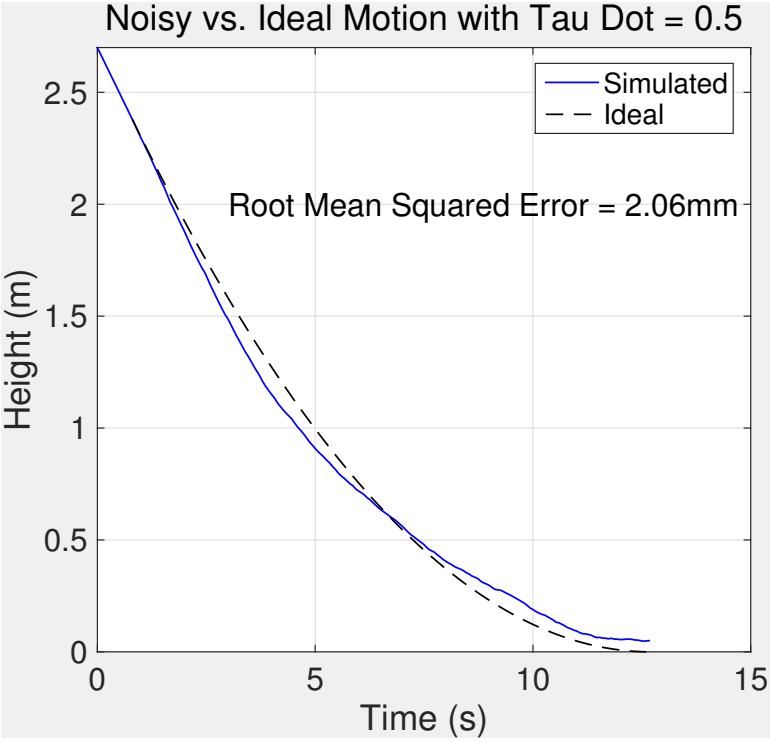

Figure 9: Simulated UAS Landing

The simulated flight path varied considerably with the seed of the pseudorandom numbers. So, to find the optimum filter specifications many simulations had to be completed. Simulations of a landing with $\dot{\tau}_r$ set to 0.50

were done using linear and quadratic filters of various buffer sizes. A thousand simulations were performed for each buffer size. A simulation with no smoothing filter was also done for comparison. The quadratic filter was postulated to give the best results because according to Equation 3 the motion for a $\dot{\tau}_r$ of 0.50 is quadratic. Indeed, the motivation for fitting polynomials as the smoothing filter was because the order of the position curve would be known by the choice of $\dot{\tau}_r$.

For each simulation the root mean squared error (RMSE) was calculated. The median of the RMSE over the 1000 trials is shown in Table 1. The best linear filter had a size of 16 and the best quadratic filter had a size of 19. The quadratic filter had the lowest RMSE as predicted. When compared to no filtering at all, both smoothing filters significantly reduced the RMSE.

Table 1: Simulation Results

	Buffer Size	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	None
Linear	Median RMSE (cm)	1.97	1.86	1.98	1.85	1.92	1.84	1.89	1.85	1.71	1.78	1.77	1.74	1.66	1.64	1.73	1.70	1.70	1.88	2.05
Quadratic	Median RMSE (cm)	2.02	1.91	2.00	1.89	1.92	1.85	1.82	1.87	1.83	1.73	1.67	1.72	1.62	1.61	1.59	1.55	1.52	1.53	2.05

4.2 Landings with a $\dot{\tau}_r$ of 0.50

The first set of UAS landings were done with $\dot{\tau}_r$ set to 0.50. For this $\dot{\tau}_r$ three groups of landings were done. One group was without a smoothing filter, another was with a size 16 linear filters, and the last was with a size 19 quadratic filter. The paths the UAS took during these landings are shown in Figure 10.

The landing maneuver was conducted seven times with no filtering, seven times with the linear filter, and twelve times with the 19 quadratic filter. The starting height of the UAS was set to 2.7m and the starting velocity was set to 0.5 m/s, but the actual starting conditions varied with each test. The dashed line in Figure 10 shows the ideal path for a $\dot{\tau}_r$ of 0.50 at the average starting height and velocity for that group of landings.

The height shown on the graph was not the true height of the UAS during a test. The UAS's was programed to perceive the ground as 0.4m closer than reality. This was to avoid the updrafts created by the UAS when it was close to

the ground. In actuality the UAS descended from 3.1m to 0.4m during these tests.

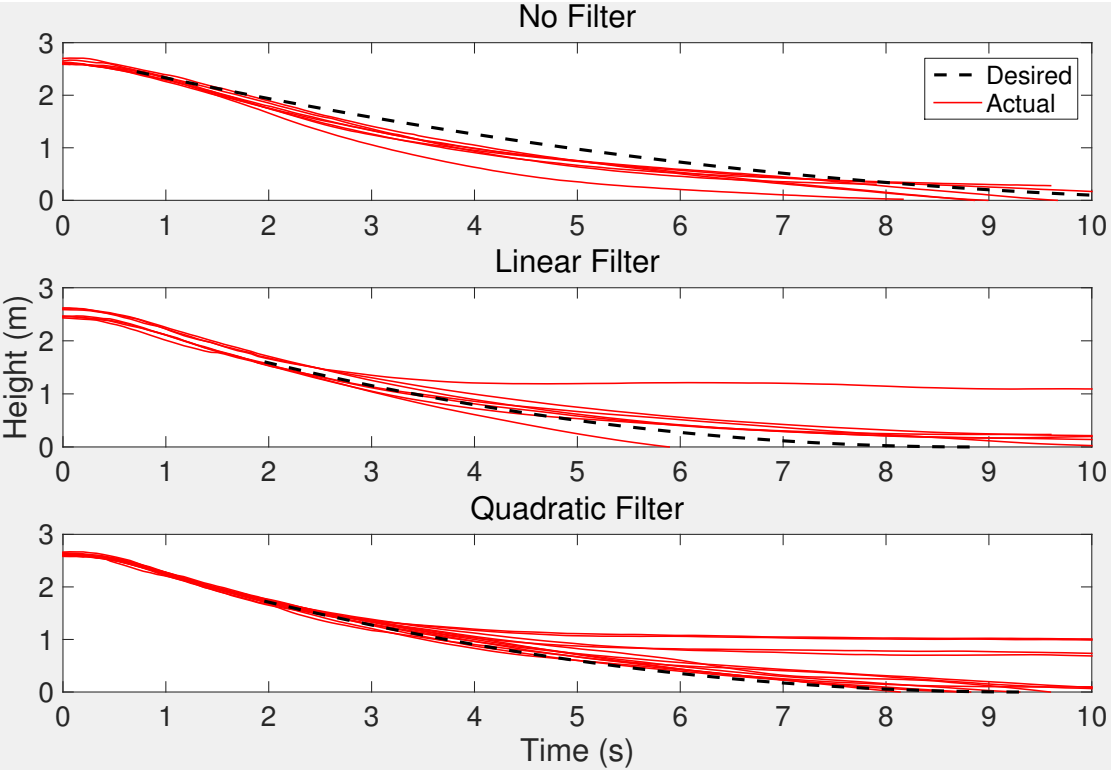

Figure 10: Actual Landings with $\dot{\tau}_r$ = 0.50

Each path shown in Figure 10 was compared with an ideal path and the RMSE between the two paths was computed. The ideal paths used for these calculations were not the dashed lines in Figure 10. The ideal path was generated using that test's actual starting conditions, not the average starting conditions. This was the same process that was done in the simulations. Table 2 displays the results.

Table 2: RMSE Analysis

	Number of Trials	Min RMSE (cm)	RMSE < 10 cm
No Filter	7	10.1	0
Linear	7	11.1	0
Quadratic	12	3.5	3

The landings that used a quadratic filter were predicted to have the highest accuracy and consistency according to the simulations. Part of this prediction ended up being true. Table 2 shows that the only landings with an RMSE less then 10 cm utilized the quadratic filter. This validates the claim that the quadratic filter had the highest accuracy. However, the quadratic filter was inconsistent in producing motions that accurate. Figure 10 shows several landings that used the quadratic filter never reached the ground, while all of the landings without a smoothing filtering followed a similar path. So while the quadratic filter had the potential to produce the most accurate landing, it was also the most inconsistent method when compared to the landings using a linear filter or no filter.

The inconsistent motions produced by the filtering process most likely had to do with an underlying assumption of fitting polynomials to a set of measurements. The filters would only produce a better true estimate of the height if the motion of the UAS were best described by quadratic or linear curve over the length of the buffer. Otherwise the smoothed estimates could be worse than the height measurements from the sensor. These poor estimates could lead to the UAS decelerating too quickly and then never reaching the ground.

In Figure 10, the landings that did not use any filter had motions all approached the ground faster than the ideal motion. This result challenged the assumption that errors in setting the UAS's velocity followed a Gaussian distribution, the assumption used in the simulations. If the velocity errors truly had a Gaussian distribution, the landing paths would be expected to deviate both above and below the ideal path.

4.2 Landings with a $\dot{t}_r$ of 0.25 and of 0.75

The next set of UAS landings were undertaken with $\dot{t}_r$ set to 0.25. Note that the filters were optimized for a $\dot{t}_r$ of 0.50. These experiments were conducted to observe how well the filters worked for different motions. For a $\dot{t}_r$ of 0.25, the order of the height versus time curve is quartic. Four landings were done without filtering, five landings were done with the linear filter, and five

landings were done with the quadratic filter. The paths the UAS took for these experiments are shown in Figure 11.

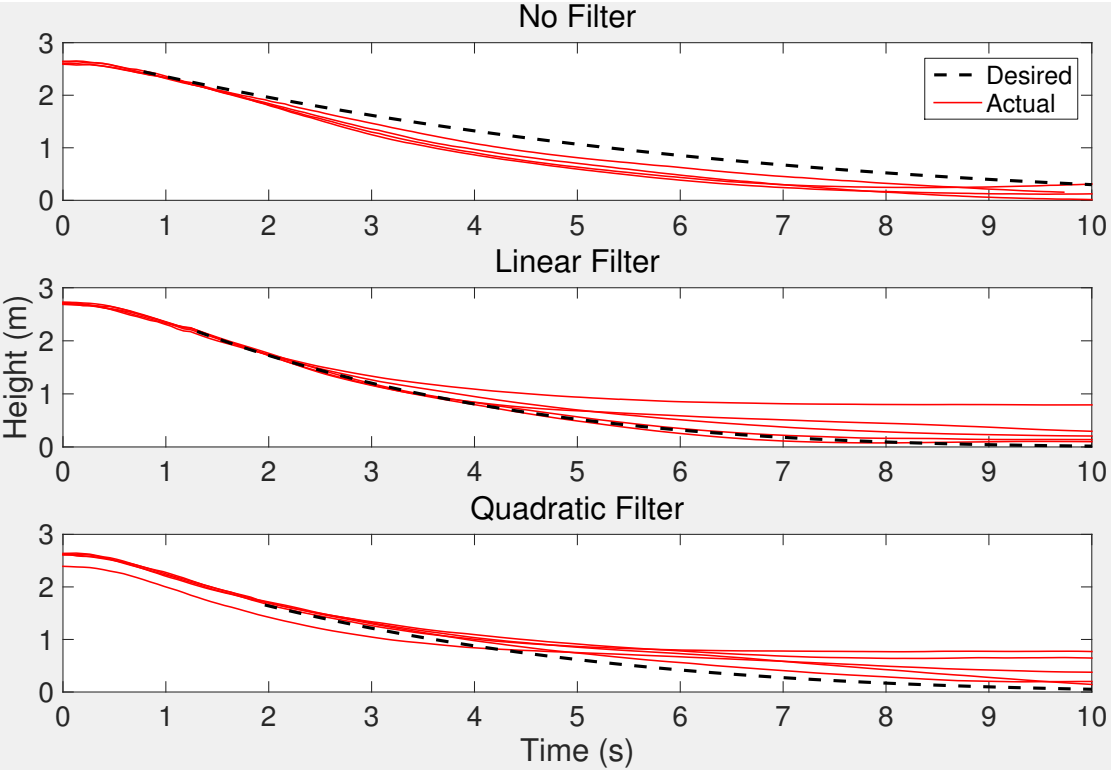

Figure 11: Actual Landings with $\dot{\tau}_r$ = 0.25

The same set of landings was done with $\dot{\tau}_r$ set to 0.75. The order of the height versus time curve for this $\dot{\tau}_r$ is 1⅓. It was expected that for this case the linear filter would perform better than the quadratic filter, since the desired motion of the UAS was closer to being linear than quadratic. The paths the UAS took for these experiments are shown in Figure 12.

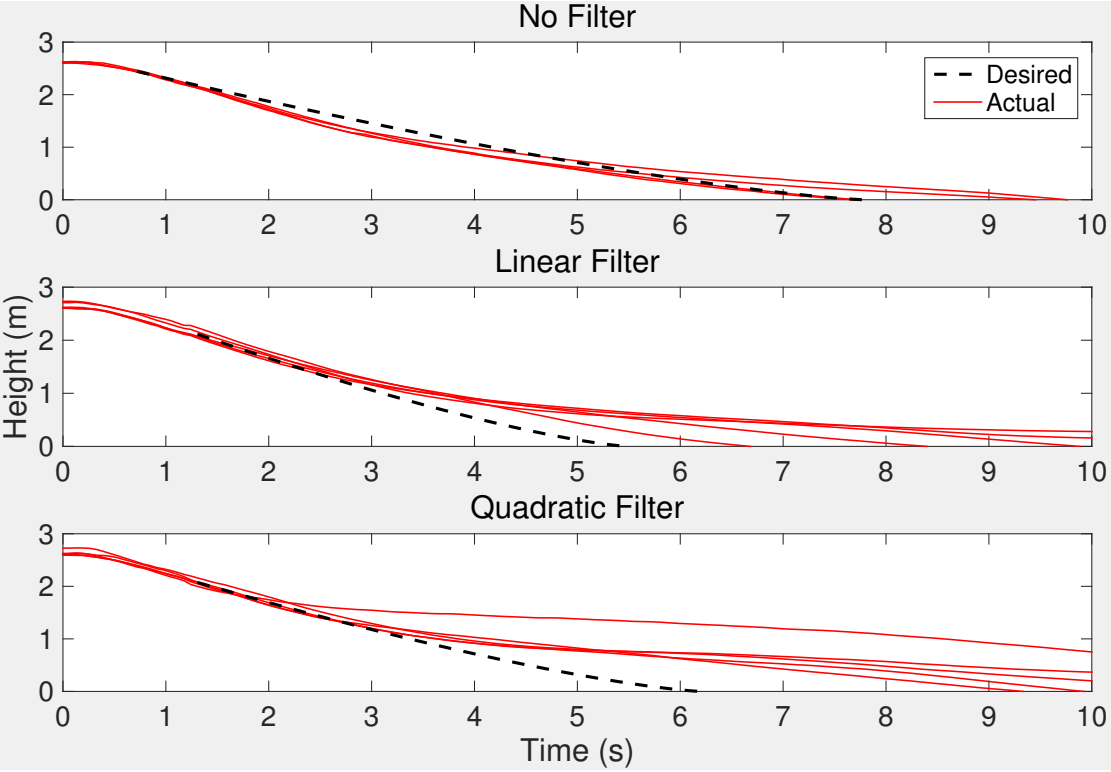

Figure 12: Actual Landings with $\dot{\tau}_r$ = 0.75

Two observations can be made from these results. The first is that the filters optimized for a $\dot{\tau}_r$ of 0.50 did not improve the accuracy of the UAS's landing when set to another $\dot{\tau}_r$. Optimizing the buffer size for the other $\dot{\tau}_r$'s could change the effects of the smoothing filters. However, it could be that the regression filters only work well for a specific range of $\dot{\tau}_r$'s.

The second observation is that the landing paths using either filter had a larger standard deviation than without any filtering. The same could be said for landings where $\dot{\tau}_r$ was 0.50. The landings using no filter did not follow the ideal motion accurately, but those landings had a smaller standard deviation. In other words, every landing done with no filter could be expected to follow a similar path even if it wasn't the desired motion.

4.3 Best Trials

Figure 13 shows the most accurate landing in a RMSE sense for each of the three $\dot{\tau}_r$'s and what filtering strategy was used.

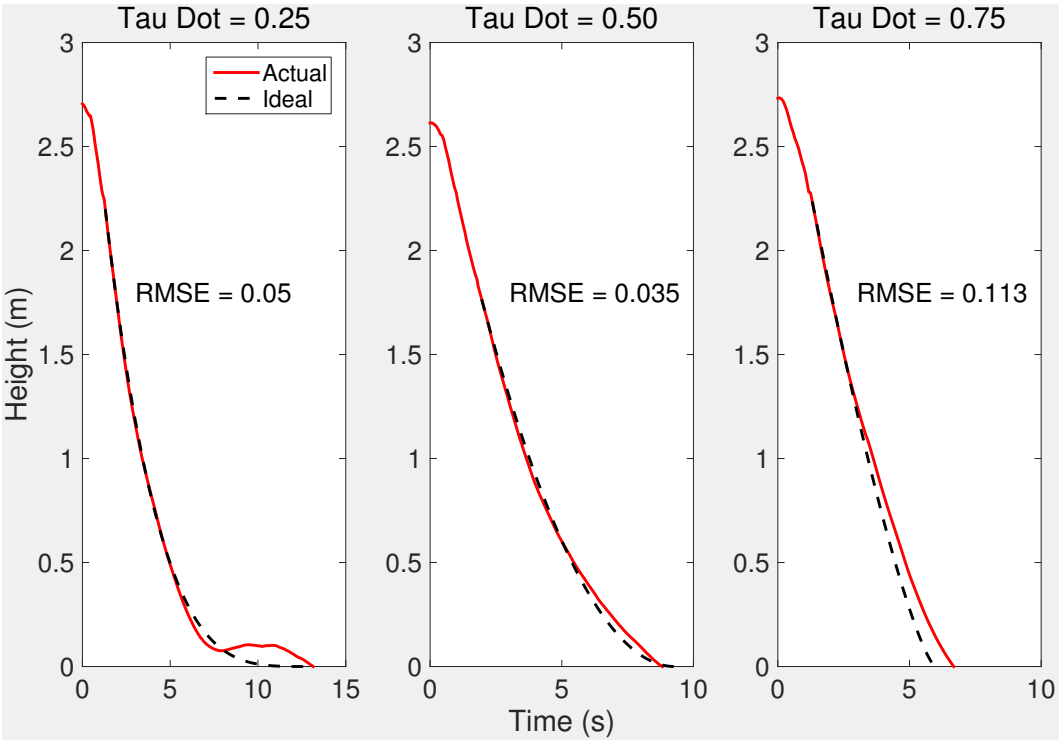

Figure 13: Best Actual Landings for Each $\dot{\tau}_r$

Figure 13 provides evidence for the conclusion that this method for landing the UAS's could produce the desired motion, even if it was inconsistent in doing so. This was especially true when $\dot{\tau}_r$ was 0.50. This case had the most accurate landing and it came from using the quadratic filter that was optimized for this $\dot{\tau}_r$ and fit polynomials of the same order as the desired motion. This supports the idea that the known order of motion could be exploited in designing a filter.

Looking at Figure 13, the $\dot{\tau}_r$ of 0.25 landing shows a large path deviation close to the ground. A smaller deviation is present in the $\dot{\tau}_r$ of 0.50 landing as well. When close to the destination, determining τ_r became difficult. This is because the velocity was close to zero near the destination. Even small errors in the measured velocity at this point would lead to large control errors.

VI. Conclusion

The landing of a UAS was controlled using echoic flow. The derivative of the time to collision, $\dot{\tau}_r$, was set to different constants during a landing to control the UAS's flight path. Height and velocity errors limited the accuracy of this method so a filter was used to smooth the height measurements. The smoothing filter sought to exploit the fact that $\dot{\tau}_r$ determines the order of the desired motion. The filter worked by fitting polynomials of the same order as the ideal motion to a buffered set of height measurements.

Three main conclusions were made from the results of using this method. The first was that this approach did yield the desired motions for the UAS's landing. The best landings are shown in Figure 13 and they show the UAS accurately following the different paths, especially for when $\dot{\tau}_r$ of 0.50. This test came from using a quadratic filter that was optimized for this case.

The second conclusion was that filtering the height estimates from the acoustic sensor could improve UAS's ability to follow the desired motions accurately. This is best shown from Table 2, where the only trials with an RMSE less than 10 cm were the ones that used a quadratic smoothing filter.

The third conclusion was that the filtering technique researched here increased the spread of landing paths taken by the UAS. Ideally the UAS would take the same path for every landing if $\dot{\tau}_r$ were not changed. This occurred because the assumption that the UAS's motion could be described by a polynomial over a large set of samples was not always accurate.

Because filtering the height estimates from the sensor did improve the UAS's landing accuracy, future work will be focused on using a different filtering algorithm. An Alpha-Beta or Kalman filter could allow for more consistency in the UAS's ability to follow the desired path. These same experiments should be conducted using those filters for a comparison. Once approaching a stationary target can be done consistently by the UAS, this research will then be used to develop a method for controlling the UAS's approach towards moving targets.

References

[1] Y. Zuqiang, F. Zhou, and L. I. Ping, “A Bio-inspired Collision-free 4D Trajectory Generation Method for Unmanned Aerial Vehicles Based on Tau Theory,” in *Proceedings of the 34th Chinese Control Conference*, 2015, pp. 6961–6968.

[2] Z. Zhang, P. Xie, and O. Ma, “Bio-inspired trajectory generation for UAV perching,” in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2013, pp. 997–1002.

[3] M. T. Alkowatly, V. M. Becerra, and W. Holderbaum, “Bioinspired Autonomous Visual Vertical Control of a Quadrotor Unmanned Aerial Vehicle,” *J. Guid. Control. Dyn.*, vol. 38, no. 2, pp. 249–262, Feb. 2015.

[4] F. Kendoul and B. Ahmed, “Bio-inspired TauPilot for automated aerial 4D docking and landing of Unmanned Aircraft Systems,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 480–487.

[5] Lee, D. N. "General Tau Theory: Evolution to Date." Perception vol. 38, pp. 837-58. 2009.

[6] C. J. Baker, G. E. Smith, A. Balleri, M. Holderied, and H. D. Griffiths, “Biomimetic Echolocation With Application to Radar and Sonar Sensing,” *Proc. IEEE*, vol. 102, no. 4, pp. 447–458, Apr. 2014.

[7] D. N. Lee, J. A. Simmons, P. A. Saillant, and F. Bouffard, ‘‘Steering by echolocation: A paradigm of ecological acoustics,’’ J. Comput. Physiol. A, vol. 176, pp. 347–354, 1995.

[8] Ackerman, E. (2015, November 4). Watch This Massive Drone Launch and Recover Another Drone in Flight. Retrieved April 6, 2016, from spectrum.ieee.org

[9] Tekla, P. (2016, January 27). Lily’s Flying Camera Drone Is Flying off Virtual Shelves. Retrieved April 6, 2016, from spectrum.ieee.org

[10] McNeal. G. (2014, July 12). FAA Approves Limited Use of Drones For Ultility Company. Retrieved April 6, 2016, from forbes.com